UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/544,492	04/07/2000	Swain W. Porter	112076-138333	1773

25943 7590 02/14/2006

SCHWABE, WILLIAMSON & WYATT, P.C.
PACWEST CENTER, SUITE 1900
1211 SW FIFTH AVENUE
PORTLAND, OR 97204

EXAMINER	
BULLOCK JR, LEWIS ALEXANDER	
ART UNIT	**PAPER NUMBER**
2195	

DATE MAILED: 02/14/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/544,492	PORTER, SWAIN W.	
	Examiner	Art Unit	
	Lewis A. Bullock, Jr.	2195	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *30 December 2005*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-29* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,4-7 and 10-29* is/are rejected.

7)☒ Claim(s) *2,3,8 and 9* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. The Office would like to make Applicant aware that a new examiner has been assigned to the application.

2. Applicant's request for reconsideration of the finality of the rejection of the last Office action is persuasive and, therefore, the finality of that action is withdrawn.

Claim Rejections - 35 USC § 112

3. The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

4. Claims 1-6, 7-12, 13-16 and 23-29 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter that was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention. The claims detail a limitation that the remapping is being performed independent of the operating system. In the paper filed April 11, 2005, Applicant state that that all amendments are fully supported by the original disclosure and that no new matter has been introduced. In that paper, there was no express indication of where the support for the limitation was disclosed. The examiner has thoroughly reviewed Applicant's originally disclosure and has found no language, either explicitly or implicitly, that would allude to the remapping being performed independent of the operating system. At best the original disclosure

states that the invention allows for the introducing of a new order of privilege relationship without requiring major re-design to the fundamental privilege protection mechanism of a processor nor the operating system that uses the processor (pg. 8, lines 1-5). This means that the remapping, thereby allows an untrustworthy program to be confined to execute out of a more privilege ring while trustworthy programs are confined to less privilege rings, without creating a backward capability problem such that the operating system has to be redesigned to allow the capability to happen (pg. 3, lines 6-20; pg. 8, lines 6-26). First, this in no way would allude to the remapping being performed totally without the use of the operating system. One of ordinary skill in the art would never come to that conclusion based upon reading the disclosure as disclosed. Independent of the operating system means that the operating system would not play any role in the remapping. The claims make no mention of whether the operating system has a role in the remapping or not. In fact, the disclosure doesn't even distinguish the operating system from the remapper. The operating system is sparingly recited throughout the specification. What is clear from the disclosure is the remapper allows for the privilege levels of untrustworthy components to be confined in the more privilege levels without any massive reconfiguring of the operating system. In addition, in order to be independent of the operating system there would have to be no re-design whatsoever to the operating system and not just precluding major redesign. The invention appears to allow some operating system redesign but not major redesign by distinguishing that that the redesign precluded is major and therefore could not show that the remapping is performed independent of the operating system since it appears

that minor redesign is allowed. Therefore, there is no support for the limitation independently of the operating system as disclosed in the claims.

In addition, claims 23-26 are not supported or enabled to operate. Independent claims 23 and 25 both detail the privilege remapper configured to remap a current privilege level using an instruction of the processor to a different current privilege level wherein the remapping is performed independent of the operating system, the instruction, and the task. It would be impossible for one to perform remapping by using an instruction and wherein the remapping is performed independent of the very instruction that is invoked to perform the remapping. In essence, you couldn't call the very instruction that performs a remap operation but somehow are capable to perform remapping.

5. Claim 5 recites the limitation "at least one of **the** one or more logical elements" in line 2. There is insufficient antecedent basis for this limitation in the claim. It appears that claim 5 should depend from claim 4, not claim 1.

Important Notation

Applicant has argued over the prior two office actions that all the claims set forth the remapper performing independently of the operating system. First, the examiner states, as outlined above, that such a limitation constitutes new matter to the original disclosure. Secondly claims 19-22 clearly teach away from such. Independent claims 19 and 21 make no mention that the remapping is performed independently and their

dependents explicitly state that the remapping is performed by the operating system. Therefore, arguments that the independent capability is not shown would be moot to these claims.

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

7. Claims 1, 4-7 and 10-29 are rejected under 35 U.S.C. 103(a) as being unpatentable over KAMIYA (U.S. Patent 4,949,238).

As to claim 1, KAMIYA teaches a processor (data processor) comprising: a control register to store a current privilege level to attribute an execution privilege level to a task for the processor (CPL of executing routine on the processor) and a privilege remapper (bus) coupled to the control register and configured to remap a current privilege level (CPL stored in CPL register) stored in the control register to a different current privilege level attributing a different execution privilege level to the task for the processor (via when segments are switched the bus reads a descriptor from the uROM or when the current privilege level register changes, a new privilege level value is supplied to and set in this current privilege level register via the bus) (col. 3, lines 20-34). However, KAMIYA does not explicitly detail that the operating system initializes the CPL. Official Notice is taken in that it is well known in the art that an operating system

can initialize a CPL value in a CPL register for a task and therefore would be obvious in view of KAMIYA in order to facilitate independent privilege checking and changing of a processor.

As to claim 7, reference is made to a method that corresponds to the processor of claim 1 and is therefore met by the rejection of claim 1 above.

As to claim 13, KAMIYA teaches on a processor, the remapping of a task from a current privilege level, i.e. first privilege level, to a new privilege level by a bus (col. 3, lines 21-44). KAMIYA also teaches a plurality of instruction segments having instructions wherein one instruction segment can transfer or invoke another instruction segment (col. 5, lines 23-35). Changes in code segments, i.e. instructions of jump or other invocation instructions, makes the system read in the new descriptor associated with the new code segment and thereby load its information, i.e. its CPL (col. 6, lines 33-41). After the loading of the CPL, a determination of memory protection violations of the code segment is checked (col. 6, lines 41-47; col. 7, lines 23-44). Official Notice is taken in that 4-level privilege systems are well known in the art. Therefore, KAMIYA teaches a sets of instructions (code segments) potentially having different current privilege levels (via the descriptors or changing by the bus) wherein a first set of instructions (code segment) is remapped to a new current execution privilege level (to a new privilege level by the bus) prior to runtime privilege checking and execution of a collection of programming instructions (via after the loading of the descriptor performing

a determination of memory protection violations) wherein the new privilege level is any of the four privilege level of a well known privilege level system. Hence, KAMIYA would teach remapping a ring-2 task to be confined to ring-3 and ring-3 task to be confined to ring-2 independently of the operating system (via the bus performing the changing of the CPL).

As to claim 28 and 29, It is well known in the art that a task is an Internet application or a task associated with an operating system.

As to claim 16, refer to claim 7 for rejection. Claim 16 further details the task being a set of instructions and there existing a second set of instructions having a different current privilege level such that the first set of instructions are remapped to a third current execution privilege level prior to runtime privilege checking and execution of a collection of programming instructions. Official Notice is taken in that 4-level privilege systems are well known in the art. KAMIYA teaches the remapping of a task from a current privilege level, i.e. first privilege level, to a new privilege level by a bus (col. 3, lines 21-44). KAMIYA also teaches a plurality of instruction segments having instructions wherein one instruction segment can transfer or invoke another instruction segment (col. 5, lines 23-35). Changes in code segments, i.e. instructions of jump or other invocation instructions, makes the system read in the new descriptor associated with the new code segment and thereby load its information, i.e. its CPL (col. 6, lines 33-41). After the loading of the CPL, a determination of memory protection violations of

the code segment is checked (col. 6, lines 41-47; col. 7, lines 23-44). Therefore, KAMIYA teaches a sets of instructions (code segments) potentially having different current privilege levels (via the descriptors or changing by the bus) wherein a first set of instructions (code segment) is remapped to a third current execution privilege level (to a new privilege level by the bus) prior to runtime privilege checking and execution of a collection of programming instructions (via after the loading of the descriptor performing a determination of memory protection violations) wherein the new privilege level is any of the four privilege level of a well known privilege level system.

As to claims 17 and 18, refer to claim 16 for rejection. Claim 17 is broader than claim 16 because claim 17 does not require that the mapping be performed independent of the operating system. However, claim 17 is substantially rejected based upon the similar reasoning as detailed regarding claim 16.

As to claims 19-22, KAMIYA teaches a method of remapping of task from a current privilege level (CPL of the executing routine for the processor) to a new privilege level prior to the execution of the task by the processor (via changes in code segments, i.e. instructions of jump or other invocation instructions, makes the system read in the new descriptor associated with the new code segment and thereby load its information, i.e. its CPL (col. 6, lines 33-41) such that after the loading of the CPL, a determination of memory protection violations of the code segment is checked (col. 6, lines 41-47; col. 7, lines 23-44)) wherein the system has a plurality of task (code segments) (col. 1, lines

13-16; col. 2, lines 67-68). It would be obvious to one of ordinary skill in the art that a program segment or microprogram having a more privilege CPL is remapped by the bus to a less privileged CPL and a microprogram having a least privileged CPL is remapped by the bus to a more privileged CPL. However, KAMIYA does not explicitly detail that the operating system initializes the CPL. Official Notice is taken in that it is well known in the art that an operating system can initialize a CPL value in a CPL register for a task and in some instances send instructions for performing the change in CPL which would either raise or lower the CPL. It would be obvious that this well known teaching in combination with the teaching of KAMIYA would allow the bus to set the CPL in response to the instruction. Therefore it would be obvious in view of KAMIYA in order to facilitate independent privilege checking and changing of a processor (via the bus).

As to claim 23, refer to claim 1 for rejection.

As to claim 25, refer to claim 1 for rejection.

As to claims 27-29, refer to claims 13-15 for rejection.

As to claims 4-6, It would be obvious to one of ordinary skill in the art that since the privilege remapper (bus) changes the CPL bits prior to runtime privilege checking (col. 3, lines 31-34) that there must exist an element for setting this new CPL. In addition, the condition in changing the CPL would be to always performing the change.

As to claims 10-12, refer to claims 4-6 for rejection.

As to claim 24, refer to claims 4-6 for rejection.

As to claims 26, refer to claims 4-6 for rejection.

Allowable Subject Matter

8. Claims 2, 3, 8 and 9 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims as well as correcting all 112 issues. The cited claims detail how the remapping alleviates major re-design of the operating system. The remapper either uses a register that stores a plurality of remapped CPLs to be accessed using the stored current privilege level or an array to store a plurality of remapped CPLs to be accessed using a configuration value and the stored current privilege levels prior to runtime privilege checking. The two cited ways allow for the remapper to change a CPL of one task to a more or less restrictive CPL. All of the cited prior art of record disclose the use of an instruction or an alternative entity that sets or changes a CPL to a new value. The cited prior art of record does not allude to mapping the new CPL value based upon the register or array as detailed in claims 2, 3, 8 or 9 and therefore these cited claims are allowable over the cited prior art of record.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Lewis A. Bullock, Jr. whose telephone number is (571) 272-3759. The examiner can normally be reached on Monday-Friday, 8:30 a.m. - 5:00 p.m..

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng An can be reached on (571) 272-3756. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

LEWIS A. BULLOCK, JR.
PRIMARY EXAMINER

February 9, 2006